

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application. The following listing provides the amended claims with the amendments marked with deleted material in [brackets] and new material underlined to show the changes made.

5 **Listing of Claims:**

1. (Currently Amended) A method for producing a circuit description of a design, the method comprising:

- a) from the design, selecting a candidate sub-network;
- b) identifying a set of output functions performed by the candidate sub-
10 network;
- c) based on the identified set of output functions, retrieving a replacement sub-network from a storage structure that stores replacement sub-networks;
- d) determining whether to replace the selected candidate sub-network with the replacement sub-network in the design;
- 15 e) if ~~determine~~ determined to replace the selected candidate sub-network, replacing the selected candidate sub-network with the replacement sub-network in the design.

2. (Original) The method of claim 1, wherein the set of output functions includes only one output function.

3. (Original) The method of claim 1, wherein the set of output functions
20 includes a plurality of output functions.

4. (Original) The method of claim 1, wherein the candidate sub-network includes multiple circuit elements.

5. (Original) The method of claim 1, wherein the storage structure stores sub-networks up to a particular complexity.

6. (Original) A method for producing a circuit description of a design, the method comprising:

a) from the design, selecting a candidate sub-network that performs a plurality output functions, wherein the candidate sub-network includes a set of circuit elements;

5 b) generating a parameter based on the output functions;

c) using the parameter to retrieve a replacement sub-network from a storage structure that stores replacement sub-networks;

d) replacing the selected candidate sub-network with the replacement sub-network in the design.

10 7. (Original) The method of claim 6 further comprising identifying the output functions performed by the sub-network.

8. (Original) The method of claim 6, wherein the set of circuit elements includes only one circuit element.

15 9. (Original) The method of claim 6, wherein the set of circuit elements includes at least two circuit elements.

10. (Original) The method of claim 6, wherein each circuit element of the sub-network has an output, and each circuit element's output provides a result of one output function performed by the selected candidate sub-network.

20 11. (Original) The method of claim 6, wherein each circuit element of the sub-network has an output, and each output function performed by the selected candidate sub-network is provided at only a circuit-element output that fans out of the sub-network.

12. (Original) The method of claim 6 further comprising:
receiving a local function for each circuit element of the selected candidate sub-network;

identifying each output function from the received local functions.

13. (Original) The method of claim 12 wherein each local or output function is represented in terms of a binary decision diagram ("BDD"), and the sub-network has at least first and second circuit elements, wherein the first circuit element performs a first local function, and the second circuit element performs a second local function, wherein the BDD of a first output function is derived from the BDD of the first local function, and the BDD of a second output function is derived from the BDD's of at least the first and second local functions.

14. (Original) The method of claim 6 further comprising
receiving the design, wherein the design is a combinational-logic network,
selecting additional candidate sub-networks;
replacing some of the selected additional sub-networks with replacement sub-networks retrieved from the storage structure;
wherein the replacement of the candidate sub-networks optimizes the combinational-logic network design.

15. (Original) The method of claim 6 further comprising
receiving a logical representation of the design;
converting the logical representation of the design to a circuit-level representation;
wherein selecting the candidate sub-network includes selecting the candidate sub-network from the circuit-level representation.

16. (Original) The method of claim 6, wherein the parameter is a set of indices for storing the sub-network in the storage structure, wherein the set of indices includes an index for each output function performed by the sub-network.

17. (Original) The method of claim 16, wherein using the parameter comprises:
a) identifying each replacement sub-network stored in the storage structure

that is associated with each index in the set of indices; and

b) retrieving each identified replacement sub-network.

18. (Original) The method of claim 16, wherein the indices are numerical indices.

19. (Original) The method of claim 18, wherein the storage structure is a

5 relational database, and the set of indices are indices into the relational database.

20. (Original) The method of claim 16, wherein the candidate sub-network has a set of input variables, wherein generating the parameter comprises:

a) identifying one of the output functions as a pivot function;

b) using the pivot function to specify a configuration for the input variables,

10 c) based on the specified input-variable configuration, specifying an index for each function.

21. (Original) The method of claim 20, wherein using the pivot function to specify an input-variable configuration comprises:

a) identifying a canonic representation of the pivot function;

15 b) selecting an input-variable configuration that results in the canonic representation of the pivot function as the specified input-variable configuration.

22. (Original) The method of claim 21 further comprising:

a) generating a truthtable representation of the pivot function;

20 b) wherein identifying the canonic representation includes identifying a canonic representation of the truthtable representation of the pivot function.

23. (Original) The method of claim 21 further comprising condensing the canonic representation to obtain a condensed representation of the pivot function.

24. (Original) The method of claim 22 further comprising using the specified input-variable configuration to specify a truthtable representation for each non-pivot function of

the sub-network, wherein the index of each non-pivot function is specified based on the truthable representation of the non-pivot function.

25. (Original) The method of claim 21 further comprising:

5 specifying a condensed representation of each non-pivot function based on the selected input-variable configuration.

26. (Original) The method of claim 20, wherein the index of the pivot function is a primary index, and the index of each non-pivot function is a secondary index, wherein using the parameter comprises:

10 a) for each particular index pair formed by the primary index and one of the secondary indices, identifying each replacement sub-network stored in the storage structure that is associated with the particular index pair;

b) determining whether any of the identified replacement sub-networks are associated with all the index pairs;

15 c) retrieving any identified replacement sub-network that is associated with all index pairs.

27. (Original) The method of claim 6 further comprising:

before replacing the candidate sub-network with the replacement sub-network, evaluating whether to replace the selected candidate sub-network with the replacement sub-network,

20 wherein said replacing is based on the evaluation.

28. (Original) The method of claim 27, wherein the evaluating comprises computing a cost function.

29. (Original) The method of claim 27 further comprising:

selecting additional candidate sub-networks;

identifying a replacement sub-network for each selected candidate sub-network;

evaluating each identified replacement;

based on the evaluations, replacing some of the candidate sub-networks with the replacement sub-networks identified for the candidate sub-networks.

5 30. (Original) The method of claim 27,

wherein retrieving the replacement sub-network comprises retrieving several replacement sub-networks, the method further comprising:

evaluating each retrieved sub-network to identify viable replacement candidates;

10 wherein the replacement sub-network that replaces the candidate sub-network is one of the viable replacement candidates.

31. (Previously Added) A computer readable medium that stores a computer program for producing a circuit description of a design, the computer program comprising sets of instructions for:

15 a) selecting, from the design, a candidate sub-network that performs a plurality output functions, wherein the candidate sub-network includes a set of circuit elements;

b) generating a parameter based on the output functions;

c) using the parameter to retrieve a replacement sub-network from a storage structure that stores replacement sub-networks;

20 d) replacing the selected candidate sub-network with the replacement sub-network in the design.

32. (Previously Added) The computer readable medium of claim 31, wherein the computer program further comprises a set of instructions for identifying the output functions performed by the sub-network.

33. (Previously Added) The computer readable medium of claim 31, wherein the

computer program further comprises sets of instructions for:

a) receiving a local function for each circuit element of the selected candidate sub-network;

b) identifying each output function from the received local functions.

5 34. (Previously Added) The computer readable medium of claim 33, wherein each local or output function is represented in terms of a binary decision diagram ("BDD"), and the sub-network has at least first and second circuit elements, wherein the first circuit element performs a first local function, and the second circuit element performs a second local function, wherein the BDD of a first output function is derived from the BDD of the first local function,
10 and the BDD of a second output function is derived from the BDD's of at least the first and second local functions.

35. (Previously Added) The computer readable medium of claim 31, wherein the parameter is a set of indices for storing the sub-network in the storage structure, wherein the set of indices includes an index for each output function performed by the sub-network.

15 36. (Previously Added) The computer readable medium of claim 35, wherein the set of instructions for using the parameter comprises sets of instructions for:

a) identifying each replacement sub-network stored in the storage structure that is associated with each index in the set of indices; and

b) retrieving each identified replacement sub-network.

20 37. (Previously Added) The computer readable medium of claim 36, wherein the candidate sub-network has a set of input variables, wherein the set of instructions for generating the parameter comprises sets of instructions for:

a) identifying one of the output functions as a pivot function;

b) using the pivot function to specify a configuration for the input variables,

c) based on the specified input-variable configuration, specifying an index for each function.

38. (Previously Added) The computer readable medium of claim 37, wherein the set of instructions for using the pivot function to specify an input-variable configuration comprises sets of instructions for:

a) identifying a canonic representation of the pivot function;

b) selecting an input-variable configuration that results in the canonic representation of the pivot function as the specified input-variable configuration.

39. (Previously Added) The computer readable medium of claim 38, wherein the computer program further comprises sets of instructions for:

a) generating a truthtable representation of the pivot function;

b) wherein identifying the canonic representation includes identifying a canonic representation of the truthtable representation of the pivot function.

40. (Previously Added) The computer readable medium of claim 39, wherein the computer program further comprises a set of instructions for condensing the canonic representation to obtain a condensed representation of the pivot function.

41. (Previously Added) The computer readable medium of claim 40, wherein the computer program further comprises a set of instructions for using the specified input-variable configuration to specify a truthtable representation for each non-pivot function of the sub-network, wherein the index of each non-pivot function is specified based on the truthtable representation of the non-pivot function.

42. (Previously Added) The computer readable medium of claim 38, wherein the computer program further comprises a set of instructions for specifying a condensed representation of each non-pivot function based on the selected input-variable configuration.

43. (Previously Added) The computer readable medium of claim 37, wherein the index of the pivot function is a primary index, and the index of each non-pivot function is a secondary index, wherein the set of instructions for using the parameter comprises sets of instructions for:

- 5 a) for each particular index pair formed by the primary index and one of the secondary indices, identifying each replacement sub-network stored in the storage structure that is associated with the particular index pair;
- b) determining whether any of the identified replacement sub-networks are associated with all the index pairs;
- 10 c) retrieving any identified replacement sub-network that is associated with all index pairs.

44. (New) A method for producing a circuit description of a design, the method comprising:

- a) from the design, selecting a candidate sub-network;
- 15 b) identifying a set of at least two output functions performed by the candidate sub-network;
- c) based on the identified set of output functions, retrieving a replacement sub-network from a storage structure that stores replacement sub-networks; and
- d) replacing the selected candidate sub-network with the replacement sub-
- 20 network in the design.

45. (New) The method of claim 44, wherein the candidate sub-network includes multiple circuit elements.

46. (New) The method of claim 44, wherein the storage structure stores sub-networks
5 up to a particular complexity.